

Plan de desarrollo

1. Página de inicio de sesión (login) que cuenta con usuario y contraseña por defecto (admin, contraseña) para acceder al sistema.
2. **Tecnología de la aplicación:** HTML, Javascript, css.
3. Entre las vulnerabilidades a subsanar tenemos:
 - Enumeración de usuarios
 - Exposición de credenciales en el código fuente.
 - Credenciales por defecto

Enumeración de Usuarios:



Página de Inicio de Sesión

Usuario:

Contraseña:

Login

Usuario incorrecto



Página de Inicio de Sesión

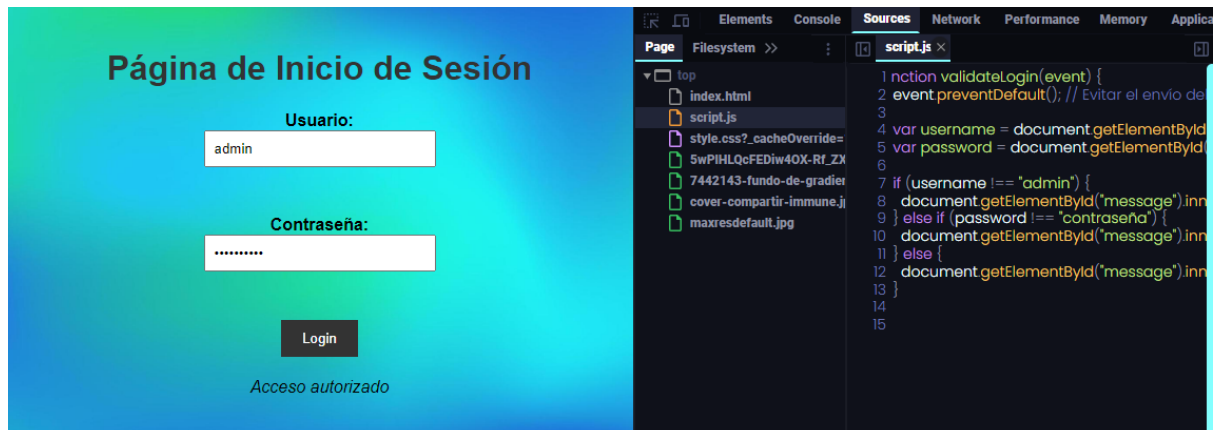
Usuario:

Contraseña:

Login

Contraseña incorrecta

Exposición de credenciales y credenciales por defecto

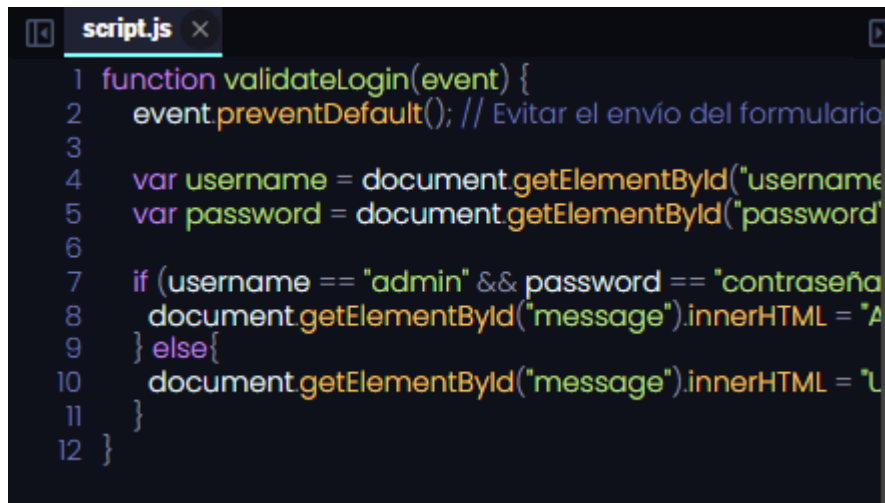


4. Vulnerabilidades ya subsanadas:



Lo que se realiza es una validación global tanto de usuario como de contraseña y se evita brindar información al atacante que pueda utilizar para enumerar usuarios del login.

Credenciales expuestas en código fuente:

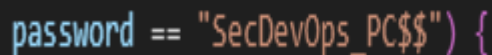


```
1 function validateLogin(event) {
2   event.preventDefault(); // Evitar el envío del formulario
3
4   var username = document.getElementById("username");
5   var password = document.getElementById("password");
6
7   if (username == "admin" && password == "contraseña") {
8     document.getElementById("message").innerHTML = "Acceso autorizado";
9   } else {
10    document.getElementById("message").innerHTML = "Acceso denegado";
11  }
12 }
```

Para evitar dicha exposición de las credenciales en el código fuente como lo observamos en la imagen se puede implementar una base de datos que guarde dichas credenciales, validarlas desde la misma y poder autorizar o denegar el acceso, de esta manera se subsanaría este fallo.

Por cuestión de tiempo no se implementa una base de datos para el proyecto (añadido a que esta es una prueba de concepto con énfasis en la enumeración de usuarios), sin embargo queda documentada su solución.

Las credenciales por defecto son más propensas a romperse en un ataque de fuerza bruta, por tanto se recomienda utilizar una contraseña más robusta (mínimo de 12 caracteres, mayúsculas, minúsculas y caracteres especiales)



```
password == "SecDevOps_PC$$") {
```